Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks

François Ingelrest, David Simplot-Ryl

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks

François Ingelrest — David Simplot-Ryl

N° 0290

Janvier 2004

THÈME 1

Rapport technique

ISSN 0249-0803

Localized Broadcast Incremental Power Protocol for Wireless Ad Hoc Networks

François Ingelrest*, David Simplot-Ryl *

Thème 1 — Réseaux et systèmes
Projet POPS

Rapport technique n° 0290 — Janvier 2004 — 15 pages

Abstract: An efficient broadcast protocol is of prime importance in a wireless ad hoc network. The best known algorithm, BIP (Broadcast Incremental Power), constructs a broadcast tree from a source node and offers very good results in terms of energy savings. Unfortunately, its computation is centralized, as the source node needs to know the entire topology of the network to compute the tree. Many localized protocols have been proposed, but none has ever reached the performances of BIP. In this paper, we propose and analyze a localized broadcasting protocol that makes use of the principles of BIP. In our method, each node is aware of the position of all its neighbors within two hops. The source applies the BIP scheme on the set of its two-hops neighbors, and includes in the message the list of its neighbors that need to retransmit, together with the desired transmission radii. Each node that receives the message with the order to relay computes the coverage of its neighborhood based on requested radii and does the same as the source node. Experimental results show that this new protocol has performances very close to other good ones for low densities, and is very energy-efficient for higher densities with performances near as good as BIP.

Key-words: Ad Hoc Networks, Energy-Efficient Broadcasting, Broadcast Incremental Power, Neighbor Elimination Scheme

* IRCICA/LIFL, Univ. Lille 1,INRIA futurs, France. Email: {Francois.Ingelrest, David.Simplot}@lifl.fr

Unité de recherche INRIA Futurs
Parc Club Orsay Université, ZAC des Vignes,
4, rue Jacques Monod, 91893 ORSAY Cedex (France)
Téléphone : +33 1 72 92 59 00 — Télécopie : +33 1 72 92 59 ??

Protocole local de diffusion à puissance incrémentale pour réseaux sans fils ad hoc

Résumé : Un protocole de diffusion efficace est de première importance dans un réseau sans fils ad hoc. Le meilleur algorithme connu, BIP (diffusion à puissance incrémentale), construit un arbre de diffusion depuis un noeud source et offre de très bons résultats d'un point de vue énergétique. Malheureusement, son calcul est centralisé, car le noeud source a besoin de connaître la topologie complète du réseau pour calculer l'arbre. Beaucoup de protocoles locaux ont été proposés, mais aucun n'a jamais atteint les performances de BIP. Dans cet article, nous proposons et analysons un protocole de diffusion local qui tire partie du principe de BIP. Dans notre méthode, chaque noeud a connaissance de la position de tous ses voisins à deux hops. Le noeud source applique BIP sur l'ensemble de ses voisins à deux hops, et inclut dans le message la liste des voisins qui doivent le retransmettre, ainsi que les rayons de transmission. Chaque noeud qui recoit le message avec l'ordre de relayer calcule la couverture de son voisinage à partir des rayons requis et fait la même chose que le noeud source. Les résultats expérimentaux montrent que ce nouveau protocole obtient des performances très proches d'autres bon protocoles pour de faibles densités, et est très efficace pour de plus hautes densitées avec des performances presque aussi bonnes que celles de BIP.

Mots-clés : Réseaux sans fils ad hoc, Diffusion économique, Diffusion à puissance incrémentale, Principe d'élimination de voisins

1 Introduction

In a network, the broadcasting task is defined to be a communication from one host to all the other ones, that is a source host decides to send a message that should be received by all hosts. In ad hoc networks, communication ranges are limited, thus many mobiles must participate to the broadcast in order to have the whole network covered. In this kind of network, this operation is useful for many things, such as route discovery, and should be as efficient as possible since it is mainly a maintenance task. The most important design criterion is obviously energy conservation, as mobiles rely upon a battery.

The easiest way to broadcast a message in such a network is known as the *Blind Flooding*, that is each node that receives the message relays it to its own neighborhood. This method obviously insures that the whole network will be covered, provided it is connected (*i.e.* there is a path between the source mobile and any other one). Unfortunately, the Blind Flooding requires *every* node to participate, leading to a lot of wasted energy, and other broadcast protocols, more efficient, must be designed.

Among the protocols that have been proposed to lessen the problem of energy consumption, many are "link-based solutions", while "node-based solutions" can offer better results. Indeed, in ad hoc networks, mobiles are generally equipped with omni-directional antennas, that is when a mobile emits a message with a given range, every of its neighbors within this range receives the message. This is known as the "Wireless Multicast Advantage" and was described by Wieselthier *et al.* [18]. They proposed a globalized heuristics known as *BIP* (*Broadcast Incremental Power*) that makes use of this and constructs an efficient broadcast tree from a source mobile to any other one.

In this paper, we propose a distributed solution based on the *BIP* algorithm. Given an initial connected graph, it allows a mobile to broadcast a message to the whole network with a low energy consumption. Its general principle is to have each node applying the *BIP* algorithm and forwarding the taken decisions with the broadcast packet. Its needs are the positions of neighbors within two hops for each node. We give experimental results that demonstrate the effectiveness of this algorithm, in terms of energy savings. Our algorithm needs a few more informations than other protocols but is able to offer better results, which are really close to the ones obtained with *BIP* and a global knowledge of the network.

The organization is as follows. We first give in the next section the needed network and energy model. Then, in sec. 3, we give a literature review of related work. In sec. 4, we present our localized version of *BIP* and discuss about coverage problems, and then we give in sec. 5 the performances obtained by simulations. We finally conclude and give direction for future work.

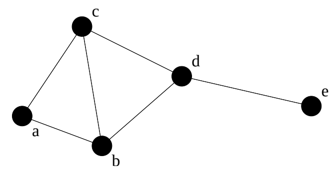


Figure 1: Distances in an ad hoc network.

2 Preliminaries

2.1 Communication Model

We represent a wireless network by a graph $G = (V, E)$ where V is the set of nodes (the mobiles) and $E \subseteq V^2$ the edge set which gives the available communications: (u, v) belongs to E means that v is a physical neighbor of u, *i.e.* u can directly send a message to v. In fact, elements of set E depend on positions and communication ranges of the nodes. Let us assume that the maximum range of communication, denoted by R, is the same for all vertices and that $d(u, v)$ is the Euclidean distance between u and v. The set E is then defined as follows:

$$E = \{(u, v) \in V^2 \mid d(u, v) \le R\}.$$

So defined graph is called the *unit graph*, with R as its transmission radius. Each node $u \in V$ must be assigned an unique value to be used as an *identifier* (*id*). We also define the neighborhood set $N(u)$ of a vertex u as:

$$N(u) = \{v \mid (u, v) \in E\}.$$

The size of this set, $|N(u)|$, is also known as the degree of u. The density of the graph is the average degree for each node. We also denote by $n = |V|$ the number of nodes in the network.

The distance between two nodes is measured in term of *number of hops*, which is simply the minimum number of links to cross from a source node to a destination one. In Fig. 1, the distance between a and b is one hop, while the distance between b and e is two hops. The one-hop neighborhood of e is $\{d\}$, while its two-hops neighborhood is $\{b, c, d\}$.

2.2 Positionning

Every broadcast protocol needs more or less informations about neighborhood of nodes. The common method used to gain this knowledge is the use of special short messages named *HELLO* messages that are periodically emitted by each node. The concept is very simple:

- each node keeps a table to store the *identifier* (*id*) of its neighbors,

- each node periodically emits a *HELLO* message containing its *id*,
- when a node u receives a *HELLO* message from a node v, it adds v to its neighborhood table, or updates the timestamp of the entry if it already exists,
- old entries are periodically deleted from the table.

When informations about the two-hops neighborhood are needed, a node can include in its *HELLO* messages its own neighborhood table, allowing its neighbors to be aware of their two-hops neighborhood. Methods exist to limit the size of the emitted packet, for instance by including only a randomly chosen part of the table instead of the whole table. This way, neighbors gradually get the whole table by storing informations obtained with each *HELLO* message.

It is also important for a node to be able to compute distances to its neighbors and between them. The simplest way to do this is to know the positions of them, by using a location system like the *GPS* [8]. If this kind of system is available, each node knows its own location and can thus include it in its *HELLO* messages. Other positioning systems or distance measurements can be found in literature [1, 2, 10].

2.3 Energy Model

In the most commonly used energy model, the measurement of the energy consumption of network interfaces when transmitting a unit message depends on the range of the emitter u:

$$E(u) = r(u)^{\alpha}, \qquad \alpha \geq 2 \tag{1}$$

where $r(u)$ is the transmitting range of u.

In reality, however, it has a constant to be added in order to take into account an overhead due to miscellaneous things such as signal processing. The general energy consumption formula becomes:

$$E(u) = \begin{cases} r(u)^{\alpha} + c & \text{if } r(u) \neq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

For instance, Rodoplu and Meng [13] consider the model with $E(u) = r(u)^4 + 10^8$, which is more realistic than the one given in Eq. 1.

3 Related work

Many solutions have been proposed to replace the inefficient Blind Flooding. Some of them only reduce the number of needed emissions to obtain a total coverage, while the others consider the possibility of radius adjustment to further reduce the energy consumption.

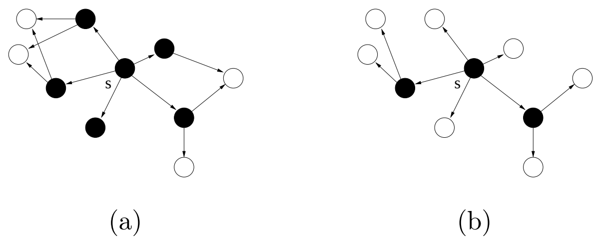


Figure 2: Applying the *MPR* algorithm.

In the first category, *MPR* (*Multipoint Relay Protocol*) has been proposed by Qayyum *et al.* [12]. It is a greedy heuristics applied by a node to compute a selection of neighbors to act as relays, in order to reach every of its two-hops neighbors. This selection is forwarded with the broadcast packet, thus slightly increasing the traffic. Fig. 2 shows an example of *MPR* relays, where s wants to broadcast a message, with black nodes being its relays. In the general case (a), each of its neighbors are relays, while in case (b), s has applied *MPR* to choose them. This protocol is very efficient in terms of energy savings and saved rebroadcast (nodes that receive the message but do not relay it).

In the same family exists the *NES* (*Neighbor Elimination Scheme*), which principle has been independently proposed in [11] and [14]. In this scheme, a node does not rebroadcast a message if all of its neighbors have been covered by previous transmissions. After each received copy of the same message, node eliminates, from its rebroadcast list, neighbors that are assumed to have correctly received the same message. If the list becomes empty before the node decides to relay the message, the rebroadcasting is canceled.

The protocol *RRS* (*RNG Relay Subset*) [3] is an improvement of *NES* where nodes limit the monitored set of neighbors to the *RNG* (*Relative Neighborhood Graph*), which was proposed by Toussaint [16]. This graph offers many advantages, like its distributed computation and its maintenance of the connectivity. Another interesting feature is its degree. Indeed, regardless of the degree of the original graph, the deduced *RNG* will have an average degree of 2.6, which is a very good characteristics (the size of the neighborhood table is greatly reduced). The protocol *RRS* has better performances than *NES* because it reduces the quantity of unnecessary transmissions, mainly due to the smaller set of monitored neighbors and to the fact that these neighbors are, in the general case, the closest ones. The protocol *RRS* is a good example of a *NES* limited to a subset of the neighborhood.

In the second category, the best known centralized algorithm is a greedy heuristics called *BIP* (*Broadcast Incremental Power*), which was proposed by Wieselthier *et al.* [18]. It is a variant of the Prim's algorithm that takes advantage of the broadcast nature of wireless transmissions. Basically, a broadcast tree is computed from a source node, by adding nodes one at time. At each step, the less expensive action to add a node is selected, either by increasing the radius of an already transmitting node, or by creating a new emission from

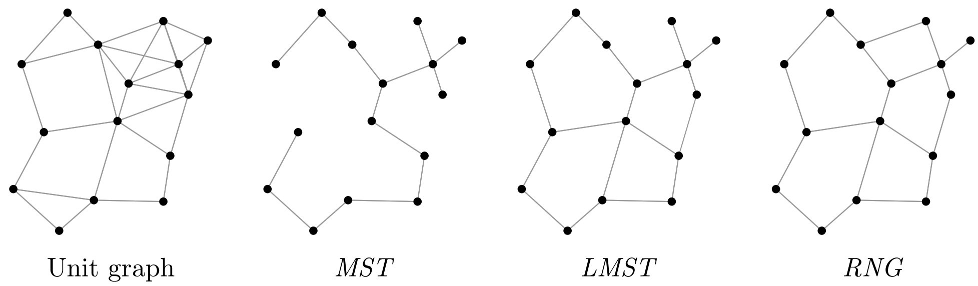
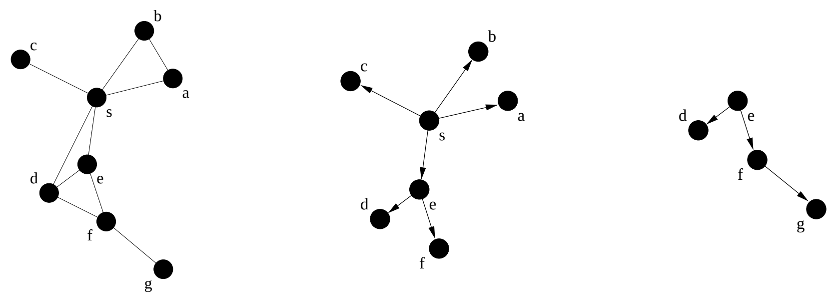


Figure 3: Example of an unit graph and its associated subgraphs.

a passive one. Although the authors considered the energy model given in Eq. 1, *BIP* fits well with the generalized energy model (Eq. 2). Indeed, this heuristics does not try to minimize nor to maximize any radius, it takes the most economical action according to the energy model, regardless of its form. Some small improvements have since been proposed but always in a centralized manner [6, 15, 17].

Wieselthier *et al.* also defined a topology control algorithm based on the *MST* (*Minimum Spanning Tree*) [18], which is used to determine the transmission range of nodes: a node selects the transmission power that permits it to cover all its neighbors in this subgraph. As, by definition, the *MST* is always connected, the graph derived from the new range assignment is also always connected. Since the computation of the *MST* is centralized, it can hardly be implemented in ad hoc networks and other protocols have been proposed, that make use of different subgraphs which can be computed locally by each node, like the *RNG* or the *LMST* (*Local Minimum Spanning Tree*) [9]. For instance, a protocol like *RBOP* (*RNG Broadcast Oriented Protocol*) [5] uses the *RNG* as a connected subgraph. This protocol can be seen as a variant of *RRS* where a transmitting node adjusts its communication radius to its furthest non-covered *RNG*-neighbor. An improvement named *LBOP* (*LMST Broadcast Oriented Protocol*) has been proposed [4] that uses the *LMST* instead of the *RNG*. The *LMST* is a localized variant of the *MST* that have been proposed by Li *et al.* [9]. Fig. 3 illustrates all these different subgraphs.

Recently, a protocol named *TR-LBOP* (*Target Radius LMST Broadcast Oriented Protocol*) based on a new concept has been proposed [7]. Most of the other protocols try to reduce the radius at each node, while it is not always an optimal behavior, due to the constant c used in the generalized energy model (Eq. 2). The protocol *TR-LBOP* uses the idea that the optimal radius should balance the values of α and c, and thus should be not too high, nor too low. This protocol offers very good results when compared to *BIP*, considering it is localized. With a density of 50, *TR-LBOP* has only an overhead of 52% when compared to *BIP*.



Base graph (a) | s' tree (b) | e's tree (c)

Figure 4: Applying *LBIP*.

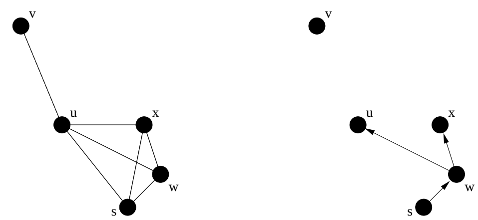
4 Localized Broadcast Incremental Power Protocol

4.1 Description of the algorithm

The goal of this protocol, referred to as *LBIP*, is to allow a local computation of a broadcasting tree by using the *BIP* algorithm. To apply it, a node needs to know the positions of their neighbors within two hops. Its principle is as follows: the source node s (the one that initiates the broadcast) computes the *BIP* tree within its two-hops neighborhood, and includes in the broadcast packet the determined radius for each of its neighbors (within two hops). When a node u receives the packet for the first time from a node v, two cases can happen:

- The packet contains some instructions for u. The relay is needed, so u constructs a *BIP* tree within its own two-hops neighborhood. This time, instead of starting from an empty tree as s did, it uses the informations contained in the packet, *i.e.* it assigns to it and its neighbors radii that were computed by v. This way, the resulting tree will not be really different from the one computed by v. Only nodes invisible for v (two-hops neighbors of u) will have to be added to the tree,
- There is no instructions for u. In this case, there is no need for u to relay the message, since v has computed a tree that covers all the neighborhood of u without its help. If every two-hops neighbors of v uses at least the computed radius, u can assume that its neighborhood will be entirely covered.

Fig. 4 shows an example of this protocol. Based on the original graph (a), the node s wants to broadcast a packet by using *LBIP*. It computes the *BIP* tree in its two-hops neighborhood, which is represented by (b). This computation allows it to detect that the node d should be covered by e instead of using a longer radius, because it is more economical.



Base graph (a) Computed tree (b)

Figure 5: Applying LBIP on one-hop neighborhood.

The chosen radius for e is written in the packet, which is sent with the radius $d(s, c)$. Nodes a, b, c receive it but do nothing, since there is no instruction for them. However, node e finds a radius for it to use, and computes the *BIP* tree based on this radius, as illustrated by (c). Similarly, f receives the packet from e and covers g by using the radius it received.

The resulting broadcast tree is obviously different than the one obtained by applying the globalized *BIP* algorithm, but we can expect that it will be similar enough to be efficient.

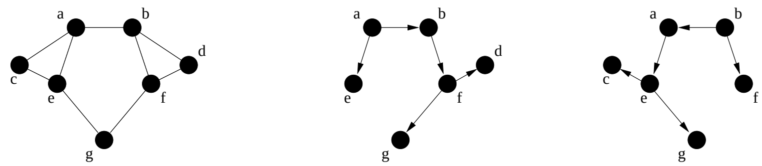
It can be noticed that this protocol is somehow similar to *MPR* [12] in its principle. When *MPR* only chooses relays for a node to reach its two-hops neighborhood, *LBIP* adds the radius adjustment dimension and forwards with the chosen nodes the determined radius.

Two behaviors can be distinguished in this algorithm. As u only knows its two-hops neighborhood, it is possible that it receives informations about unknown nodes to it (some two-hops neighbors of v are three-hops neighbors of u). While the node u cannot use these informations about these too distant nodes, it can leave them in the packet for other nodes that would be concerned.

Indeed, depending on the computed tree, it is possible that a node receives information about unknown nodes, that will be useful for its neighbors. In this case, if those informations are left in the packet, we can expect better results since the next computed trees (by next nodes) will take advantage of these informations. However, this leads to an increase of the packet size, that would result in more collisions at the *MAC* layer and an higher latency.

4.2 Coverage discussion

We discuss here of the relevance of choosing one-hop instead of two-hops neighborhood. Indeed, it seems at first that this algorithm could be applied only on the physical (one-hop) neighborhood, which requires less informations. Considering Fig. 5, we observe on the base graph (a) that u is the only neighbor of s able to reach v, but if s only knows its physical neighborhood, it is not aware about this. So the computed *BIP* tree, illustrated by (b), will



Base graph (a) a's tree (b) b's tree (c)

Figure 6: Conflicting decisions made by nodes a and b.

not instruct u to join v. So when u receives the message, it does nothing, v is not covered and the coverage is not total.

This means that each node that has non-common neighbors with the emitter (nearby all nodes in the network) must construct its own tree, regardless of the received informations, leading to an inefficient broadcast. Indeed, the *BIP* algorithm starts by creating the smallest possible link from the node, meaning that almost every node will have to emit the message once. This behavior is clearly suboptimal and ineffective.

Due to the distributed aspect of this algorithm, it is sometimes possible that two nodes take different decisions, resulting in a break in the diffusion. This case is illustrated by Fig. 6, where (a) is the base graph. If we consider that c has already got the message and that a is aware about this (the message was received from a common neighbor, not represented on the graph), the constructed tree from the point of view of a will be as shown by case (b), that is e receives the message and is instructed to not relay it. Symmetrically, if b knows that d has already received the message, its tree will be as shown by case (c). Node a computes a tree that covers d, while b does not, leading to the choice of f to act as relay to join g. Node b computes a tree that covers c while a does not, and chooses e to cover g. The two trees conflict with each other, and g is not covered.

To avoid this situation, we use the principle of *NES*, that is each node that receives the message starts monitoring its neighborhood, regardless of the decision it took (relay or not). If after a given timeout it appears that some neighbors could have not received the message, then the node sends it to them. This way, we insure that the coverage is total, at the cost of a few useless emissions.

Indeed, it is possible that a node thinks one of its neighbors has not received the message, while it is not the case, leading to an unnecessary rebroadcast. To limit this, it is possible to reduce the monitored neighbors to a smaller subset of the neighborhood, like *RRS* does, by using a connected subgraph like the *RNG* or the *LMST* one. The latter is a good choice, since it has a lower degree than *RNG* and keeps the smallest edges, but has unfortunately a higher complexity of computation. In the next section we give the variation of performances depending on the chosen subgraph.

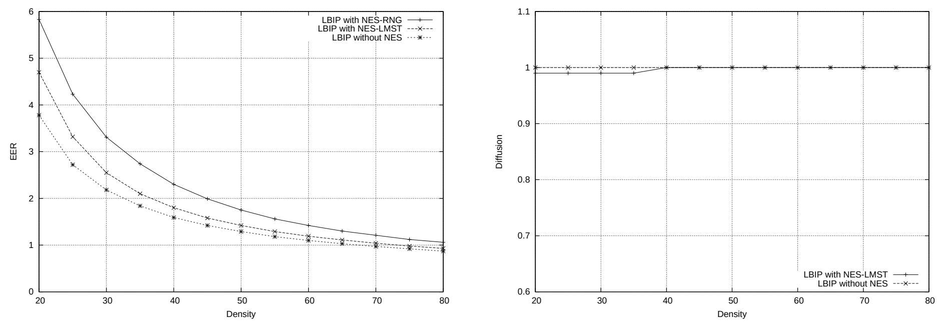


Figure 7: Efficiency of the *NES*.

5 Performances

In our simulations, we compare *LBIP* with *BIP* and *TR-LBOP*, since these two protocols are very effective in reducing the energy consumption by adjusting needed radii. We use the energy model proposed by Rodoplu and Meng [13], that is the power consumption of an emission with a radius r is given by:

$$PC(r) = r^4 + 10^8.$$

The parameters of our simulations are the following. The network is static and is always composed of 300 nodes randomly placed in a square area whose size is computed to obtain a given density. The initial maximum communication radius R is fixed to 250 meters. The timeout used in the neighbor elimination scheme is randomly generated. For each measure, 500 broadcasts are launched and for each broadcast, a new connected network is generated.

To compare the different protocols, we observe the total power consumption over the network when a broadcast has occurred. We compute a ratio named *EER*, that represents the energy consumption of the considered protocol compared to the energy that would have been spent by a Blind Flooding (each node retransmits once with the maximum radius R). The value of *EER* is so defined by:

$$EER = \frac{E_{protocol}}{E_{flooding}} \times 100.$$

We also observe the value of the *SRB* (*Saved Rebroadcast*) which is the percentage of nodes in the network that received the message but did not relay it. A Blind Flooding has a *SRB* of 0%, since each node has to retransmit once the message.

Fig. 7 shows the performance issues of *LBIP* used with or without a *NES*. The left graph illustrates the variations of efficiency of *LBIP* when using a *NES* in combination with two different subgraphs. The *MAC* layer is assumed to be ideal, that is no collisions occur when

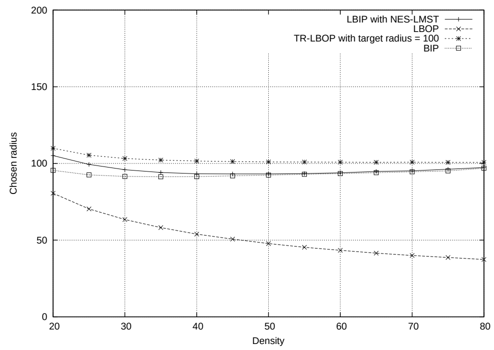


Figure 8: Average chosen radius.

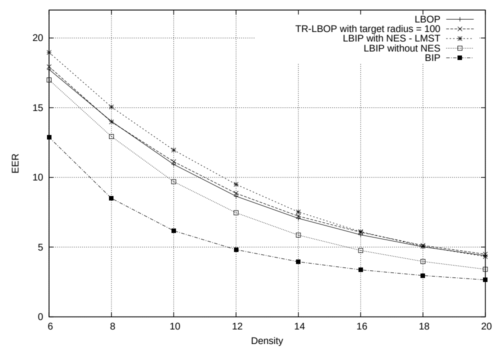


Figure 9: Efficiency for low densities.

two neighbors emit simultaneously. As expected, the use of *NES* reduces the performances, since some nodes assume there are uncovered neighbors, while they were actually covered, and rebroadcast needlessly. At this cost however, we insure that no uncovered node exists. This graph also shows that at the cost of a more complex computation, the *LMST* offers better results, which seems logical since it has been demonstrated to be a subgraph of the *RNG* [4].

The right graph shows the reachability of *LBIP* with or without a *NES*. As stated in Sec. 4.2, even in an ideal environment some contradictory decisions can happen, leading to a partial coverage of the network. However, this does not happen too often, and the reachability stay at a very acceptable level. From the density of 35, the number of non-covered nodes is so small that the reachability is virtually total.

Fig. 8 gives the average chosen radius by each node for the different used algorithms. It can be observed that the average radius for *TR-LBOP* is roughly equal to 100 meters, which seems logical since it is the distance used as the target radius. An interesting thing to note is the average radius of *BIP* and *LBIP*, which is also very near from 100 meters. As these algorithms do not use this value in their computation, it seems to confirm the theory of *TR-LBOP* that uses this value as the optimal radius for this energy model. The average radius of *LBOP* is very low, since it makes nodes cover their *LMST*-neighbors which are the nearest ones.

Performances for low densities are given in Fig. 9. It can be observed that *LBIP* offers performances very close to the other protocols. The version without a *NES* is even better than the others for all the ranges, while offering a little less insurance in the reachability, as stated by Fig. 7.

Fig. 10 illustrates the performances of *LBIP* when compared to *BIP* and *TR-LBOP*. Again, the *MAC* layer is assumed to be ideal. The protocol *TR-LBOP* has been applied when considering a target radius equal to 100, which is cited as the optimal radius with the considered energy model [7]. It can be noticed that *LBIP* has a lower energy consumption

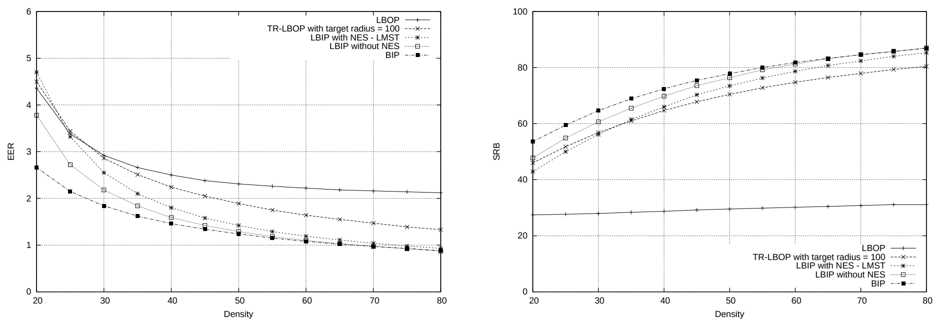


Figure 10: Efficiency of *LBIP*.

than *TR-LBOP* and a very small overhead compared to *BIP*. At a large density like 80, this overhead is virtually equal to zero, while the overhead of *TR-LBOP* is still around 63%. The value of the *SRB* is also very good for *LBIP* and approaches the performances of *BIP*. It can be observed that *LBIP* always has a greater number of passive nodes than *TR-LBOP*, which explains why the energy consumption is always lower, their chosen radii being nearly the same.

Fig. 11 gives the latency (*i.e.* the elapsed time between the launch of the broadcast and its end), where each protocol has been used with the same parameters for the *NES*. Obviously, *LBIP* has a much lower latency than the others, which can be explained by the instant retransmission of nodes, if it is needed. The *NES* is only applied after a decision has been taken, while the other protocols apply it before taking any decision, leading to an increase in the needed time for the broadcast. A low latency allows the protocol to be more

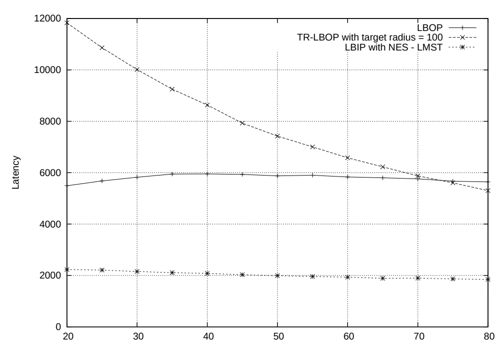


Figure 11: Latency of *LBIP*.

efficient, considering the possible mobility of nodes, and reduce the probability of collision with other communications.

6 Conclusion

In this paper, we presented a new broadcast protocol that uses the principles of *BIP* in a local way. At the cost of a few more informations stored in the broadcast packets, when compared to other protocols, *LBIP* offers very good results. Its drawback is a larger required knowledge, compared to *TR-LBOP* which requires only one-hop knowledge. However this requirement brings results really close to *BIP* which still requires a global knowledge of the network to achieve this. With somewhat less informations, our protocol obtains close performances.

As future work, we want to consider the performances of *LBIP* in a more realistic environment, and thus do some experiments with a real *MAC* layer. Indeed, as the value of the *SRB* is relatively high, we think that this protocol can offer good results when the network is faced to a high load. A high number of non-retransmitting nodes means a lower traffic.

References

[1] P. Bahl and V.N. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *Proceedings of the IEEE Infocom 2000*, Tel Aviv, Israel, March 2000.

[2] A. Benlarbi-Delaï, D. Simplot, J. Cartigny, and J.-C. Cousin. Using 3D indoor microwave phase sensitive stereoscopic location system to reduce energy consumption in wireless ad hoc networks. In *Proceedings of the Smart Objects Conference (sOc'2003)*, Grenoble, France, May 2003.

[3] J. Cartigny, F. Ingelrest, and D. Simplot. RNG relay subset flooding protocols in mobile ad hoc networks. *International Journal of Foundations of Computer Science (IJFCS)*, 14(2):253–265, April 2003.

[4] J. Cartigny, F. Ingelrest, D. Simplot-Ryl, and I. Stojmenović. Localized LMST and RNG based minimum-energy broadcast protocols in ad hoc networks. *Ad hoc Networks*, 2004. To appear.

[5] J. Cartigny, D. Simplot, and I. Stojmenović. Localized minimum-energy broadcasting in ad hoc networks. In *Proceedings of the IEEE Infocom 2003*, San Francisco, USA, April 2003.

[6] R.J. Marks II, A.K. Das, M. El-Sharkawi, P. Arabshahi, and A. Gray. Minimum power broadcast trees for wireless networks: Optimizing using the viability lemma. In *Pro-*

ceedings of the International Symposium on Circuits and Systems (ISCAS 2002), pages 245–248, Scottsdale, USA, 2002.

[7] F. Ingelrest, D. Simplot, and I. Stojmenović. Target transmission radius over LMST for energy efficient broadcast protocol in ad hoc networks. In *Proceedings of the IEEE International Conference on Communications (ICC'04)*, Paris, France, June 2004.

[8] E.D. Kaplan. *Understanding GPS: Principles and Applications*. Artech House, 1996.

[9] N. Li, J. C. Hou, and L. Sha. Design and analysis of an MST-based topology control algorithm. In *Proceedings of the IEEE Infocom 2003*, San Francisco, USA, April 2003.

[10] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AOA. In *Proceedings of the IEEE Infocom 2003*, San Francisco, USA, April 2003.

[11] W. Peng and X.C. Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In *ACM MobiHoc 2000*, pages 129 – 130, Boston, Massachusetts, USA, August 2000.

[12] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *Proceedings of the Hawaii International Conference on System Sciences (HICSS'02)*, January 2002.

[13] V. Rodoplu and T. H. Meng. Minimum energy mobile wireless networks. In *Proceedings of the IEEE International Conference on Communications (ICC'98)*, volume 3, pages 1633 – 1639, Atlanta, USA, June 1998.

[14] I. Stojmenović and M. Seddigh. Broadcasting algorithms in wireless networks. In *Proceedings of the International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet SSGRR*, L'Aquila, Italy, July 31-Aug. 6 2000.

[15] T.Chu and I.Nikolaidis. Energy efficient broadcast in mobile ad hoc networks. In *Proceedings of the International Conference on Ad-Hoc, Mobile, and Wireless Networks (ADHOC-NOW'02)*, pages 177–190, Toronto, Canada, 2002.

[16] G. Toussaint. The relative neighborhood graph of finite planar set. *Pattern Recognition*, 12(4):261–268, 1980.

[17] P.-J. Wan, G. Calinescu, X.-Y. Li, and O. Frieder. Minimum energy broadcast routing in static ad hoc wireless networks. *ACM/Kluwer Wireless Networks*, 2002.

[18] J. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proceedings of the IEEE Infocom 2000*, pages 585 – 594, Tel Aviv, Israel, 2000.

Unité de recherche INRIA Futurs
Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-0803